sensations for a force feedback system. Embodiments are disclosed which provide for efficient management of device memory and force effect output.

[0011] More specifically, the present invention provides force effects with a force feedback device having local control of the output of force sensations. A representation of device memory is created, where the device memory is provided on the force feedback device, and the representation is allocated in memory of the host computer. A force effect load command is received from an application program running on the host, where the force effect load command instructs that data for a force effect be stored in the device memory. It is then determined whether the device memory can store the force effect by examining the representation of device memory. If the device memory can store the force effect, the data for the force effect is send to the force feedback device to be stored in the device memory. The force feedback device uses the data to control a force output to a user of the force feedback device. If the device memory can store the force effect, the data for said force effect can be stored in a location of the representation of device memory, or an indication of the memory space taken by the force effect can be stored. The application program can later send a play command or a stop command, which is sent to the force feedback device to output or stop the output of force based on the loaded force effect. The total number of playing force effects loaded to the device are summed to provide the total output force on the user.

[0012] In a different aspect of the present invention, the management of storage of force effects in a force feedback system includes receiving a force effect create command by a driver running on the host computer. The command is sent from an application program running on the host computer and instructs that particular force effect data for a particular force effect be stored in memory local to the force feedback device. It is then determined whether the local memory has sufficient space to store the particular force effect data. If there is sufficient space, the particular force effect data is sent to the force feedback device to be stored in the local memory. If there is not sufficient space, the particular force effect data is stored in a cache implemented in host computer memory instead of the local memory. When a command is later received by the driver to output the cached force effect to a user of the force feedback device, the driver swaps the particular force effect data with loaded force effect data in the local memory and instructs the force feedback device to output the particular force effect.

[0013] Preferably, the driver creates a representation of the local memory in the host computer memory, and the representation can be examined for sufficient space for the force effect. Alternatively, the force feedback device can be queried and a response can be received indicating whether sufficient space is available. In addition, it can be determined whether a force effect can be loaded by comparing a priority of the particular force effect with a priority of one or more loaded force effects, where the greater priority effect can be loaded to the device memory. Priority of the loaded force effect(s) can be determined based at least partially on whether the loaded force effect is currently being output by the device, on the time period since the loaded force effect was last output by said device, and/or on whether said loaded force effect is likely to be output based on a direction of movement of a manipulandum of the force feedback device in a workspace of the manipulandum of the device. The priority can also be predefined, e.g. by the application program. Furthermore, force effects can be grouped in categories to help determine which loaded force effects can be swapped with cached force effects. An apparatus for managing storage of effect using a host cache operates as described above.

[0014] In a different aspect of the present invention, forces are output from a force feedback device coupled to a host computer. A force effect play command is received from the host computer which instructs that a particular force effect be output by the force feedback device. The data for the particular force effect and data for at least one other force effect is stored in a memory local to the force feedback device. An identification of the particular force effect is designated in a playlist in local memory. When a force is to be output, the playlist is examined to determine which of the stored force effects are to be output. A force is then determined based on the force effects designated in the playlist and the force is output to a user of the force feedback device. Preferably, the output force is based on a sum of contributions from the force effects designated in the playlist. A number can be stored in local memory indicating how many the force effects stored in local memory are currently designated to be output. This allows efficient access to only the playing force effects on the device.

[0015] In yet another aspect of the present invention, force output is provided to a user of a force feedback device only at predetermined time intervals. A first force to be output by actuators of the force feedback device is determined and then output at a first point in time occurring when a predetermined time interval has passed. A second force to be output is then determined. If the predetermined time interval has not passed when the second force has been determined, then the device waits for a second time interval and outputs the second force at a second point in time. If the predetermined time interval has passed when the second force has been determined, indicating the processing of the force has taken longer than one time interval, then the device waits for a successive time interval after an integer number of the predetermined time intervals has passed, and outputs a third force at the successive point in time. The third force is appropriate to the successive point in time. For example, the first force and the second force can be at least partially based on a periodic function that varies with time, so that the third force is based on an appropriate later point of the periodic function. This allows a small time interval to be used and thus faster updating of output forces; during infrequent intervals where force processing takes longer than one time interval, the force can be output at later intervals.

[0016] The present invention provides several embodiments for managing force effect and force output in a force feedback system. A representation of the device memory is preferably maintained in host computer memory to allow the host computer to efficiently determine when effects can be loaded in device memory. Host caching of force effects allows the application program to function as if the device can store an almost unlimited number of effects, thereby freeing the application from managing low-level processing and swapping of force effects. The playlist and discrete interval force output on the force feedback device allows efficient and high fidelity output of force sensations.